

IN THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Please cancel claims 1-10.

~~Please add the following new claims:~~

11. (New) A system comprising:

a portal server to retrieve classfiles from an Internet site on behalf of a data processing device, the data processing device comprising a processor for processing program code and an interpreter module for interpreting classfiles,

wherein the portal server comprises a content conversion module to analyze and convert the classfiles prior to transmission to the data processing device, the conversion comprising rearranging elements of two or more classfiles to form a unified programming object, the elements including a plurality of constant pool entries and one or more methods and/or fields; and

wherein after the portal server generates the unified programming object, it transmits the unified programming object to the data processing device.

12. (New) The system as in claim 11 wherein, to rearrange the elements of the two or more classfiles, the content conversion module combines redundant constant pool entries from said two or more classfiles to form a global constant pool entry in a shared constant pool within said unified programming object.

13. (New) The system as in claim 12 wherein combining further comprises:

rewriting said global constant pool entries to point to elements contained within said unified programming object, said elements corresponding to elements contained in said one or more class files and previously identified by said one or more redundant constant pool entries.

14. (New) The system as in claim 13 wherein one of said global constant pool entries is a methodref entry and said element identified by constant pool entry is a method copied to said unified programming object from said one or more class files.

15. (New) The system as in claim 14 wherein, to rearrange the elements of the two or more classfiles, the content conversion module converts numeric references to local entries within a bytecode in said method to pointers to global constant pool entries.

16. (New) The system as in claim 15 wherein, to rearrange the elements of the two or more classfiles, the content conversion module converts an exception table associated with said method to references to jop objects instead of numeric references to addresses of bytecodes.

17. (New) The system as in claim 13 wherein one of said global constant pool entries is a fieldref entry and said element identified by constant pool entry is a field copied to said unified programming object from said one or more class files.

18. (New) The system as in claim 11 wherein, to rearrange the elements of the two or more classfiles, the content conversion module validates said two or

more class files before mapping said elements to form said unified programming object.

19. (New) The system as in claim 18 wherein, to rearrange the elements of the two or more classfiles, the content conversion module converts the classfiles into a graph of jop objects to track where jump operations pointed before modification of said classfiles; adjusts constant pool references from local to global numbers based on said graph; and combines the classfiles into the unified programming object.
